Harvesting RDF metadata

Building digital library portals with harvested metadata workshop

EU-DL All Projects concertation meeting
DELOS Network of Excellence
Standardisation Forum -February 2001

Manjula Patel

UKOLN

University of Bath

Bath, BA2 7AY

Email

m.patel@ukoln.ac.uk

URL

http://www.ukoln.ac.uk/

UKOLN is funded by Resource: The Council for Museums, Archives and Libraries, the Joint Information Systems Committee (JISC) of the Higher Education Funding Councils, as well as by project funding from the JISC and the European Union. UKOLN also receives support from the University of Bath where it is based.

Outline

- Namespaces and application profiles
- SCHEMAS registry
- How to register a schema/application profile?
- Preparing your application profile for registration
- SCHEMAS application profile format
- Conclusions

Namespaces and application profiles

Namespaces:

Declare names and definitions of vocabulary terms

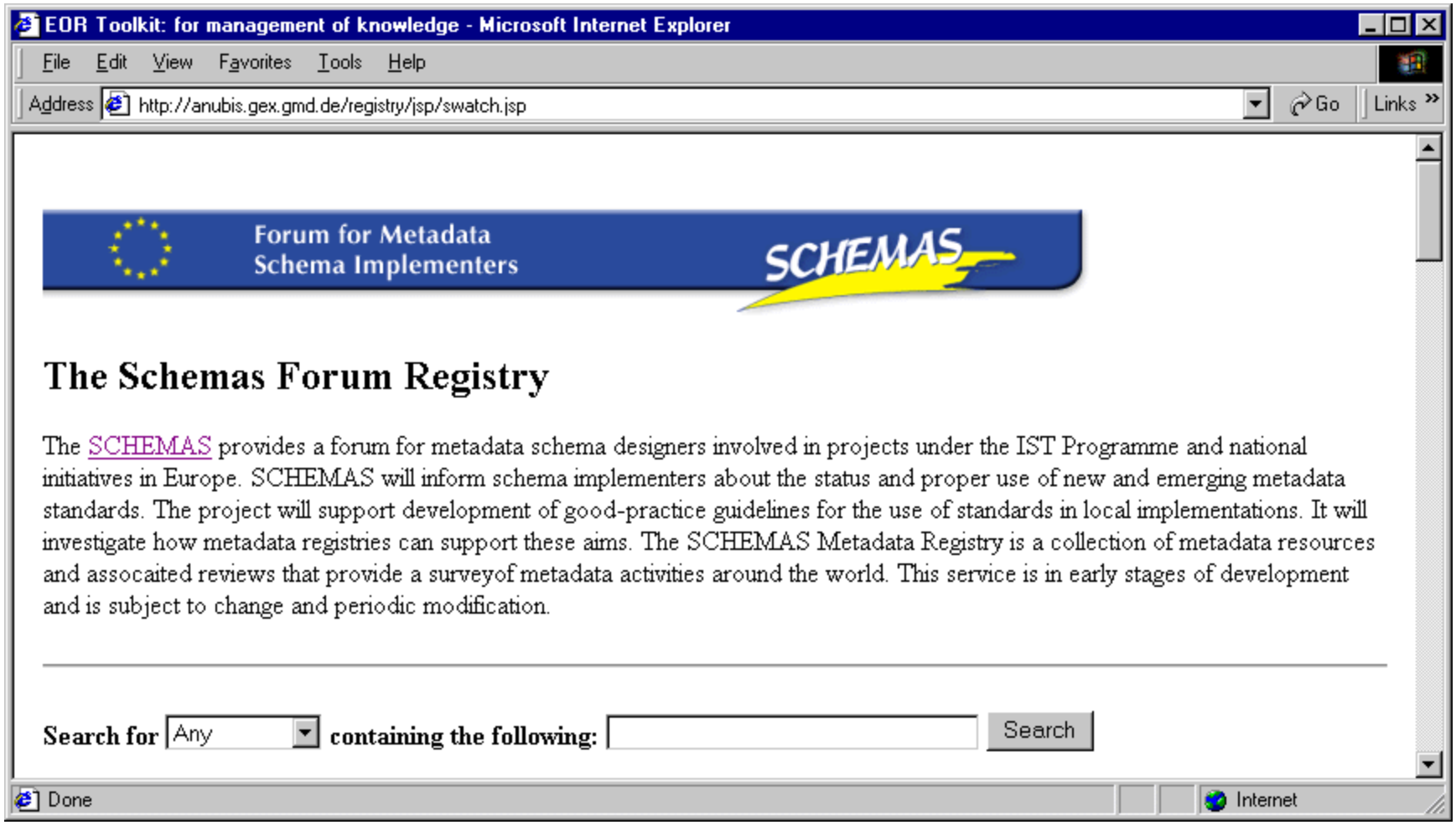
(e.g. Dublin Core standard, LCSH controlled vocabulary)

Application profiles (APs):

Consist of data elements drawn from one or more namespaces, optimised for a particular local application

- need to declare which elements are being used
- may specify dependencies e.g. mandate schemes
- may adapt existing definitions for local purposes
- may declare rules for content (usage guidelines)
- may specify whether an element is mandatory, optional or repeatable

SCHEMAS Registry



EOR Toolkit

- EOR (Extensible Open RDF) Toolkit: (http://eor.dublincore.org - *Eric Miller, OCLC*)
- Harvests RDF schemas from distributed servers on the web
- Creates central index for searching
- Schema browser -hyper-links not only between vocabularies, but between related terms

RDF Schemas

- W3C Candidate Recommendation **(March 2000)**
- Rich, web-based publication format for declaring semantics (XML for exchange)
- Capability to explicitly declare semantic relations between vocabulary terms
- Machine readable, but also defines properties and classes with human readable labels and comments

Registering an AP

For applications with a simple data model :

- submit a URL to an RDF/XML encoding
- fill in template files and submit to registry
- fill in a web form that generates appropriate RDF/XML

For applications with complex data models:

For the moment, point to a web-page with a human-readable schema

Preparations for registration ...

- **which vocabulary terms are new**
- **which elements reference other namespace**
- **honour elements predefined in other namespaces**
- **which elements need a modified definition/label**
- **which encoding schemes are required**
- **refinements of existing vocabulary terms**
- **which elements mandate a scheme**
- **mandatory, optional or repeatable elements**

Example: Renardus

Renardus metadata set (based on v0.1 draft)

Dublin Core element set (dc):

Title, Creator, Description, Subject, Identifier, Language, Type

Dublin Core element qualifiers (dcq):

Alternative (*Title*)

Encoding schemes (dc):

Subject, Type

New elements (rmes):

Country, Full Record URL, SBIG ID

Example: Renardus

New elements, qualifiers and schemes would go into a Renardus namespace schema (rmes)

Renardus application profile schema collects together usage declarations

i.e. re-using elements, qualifiers and schemes from dc, dcq and rmes.

XML namespace mechanism

...used to pull in namespaces ...

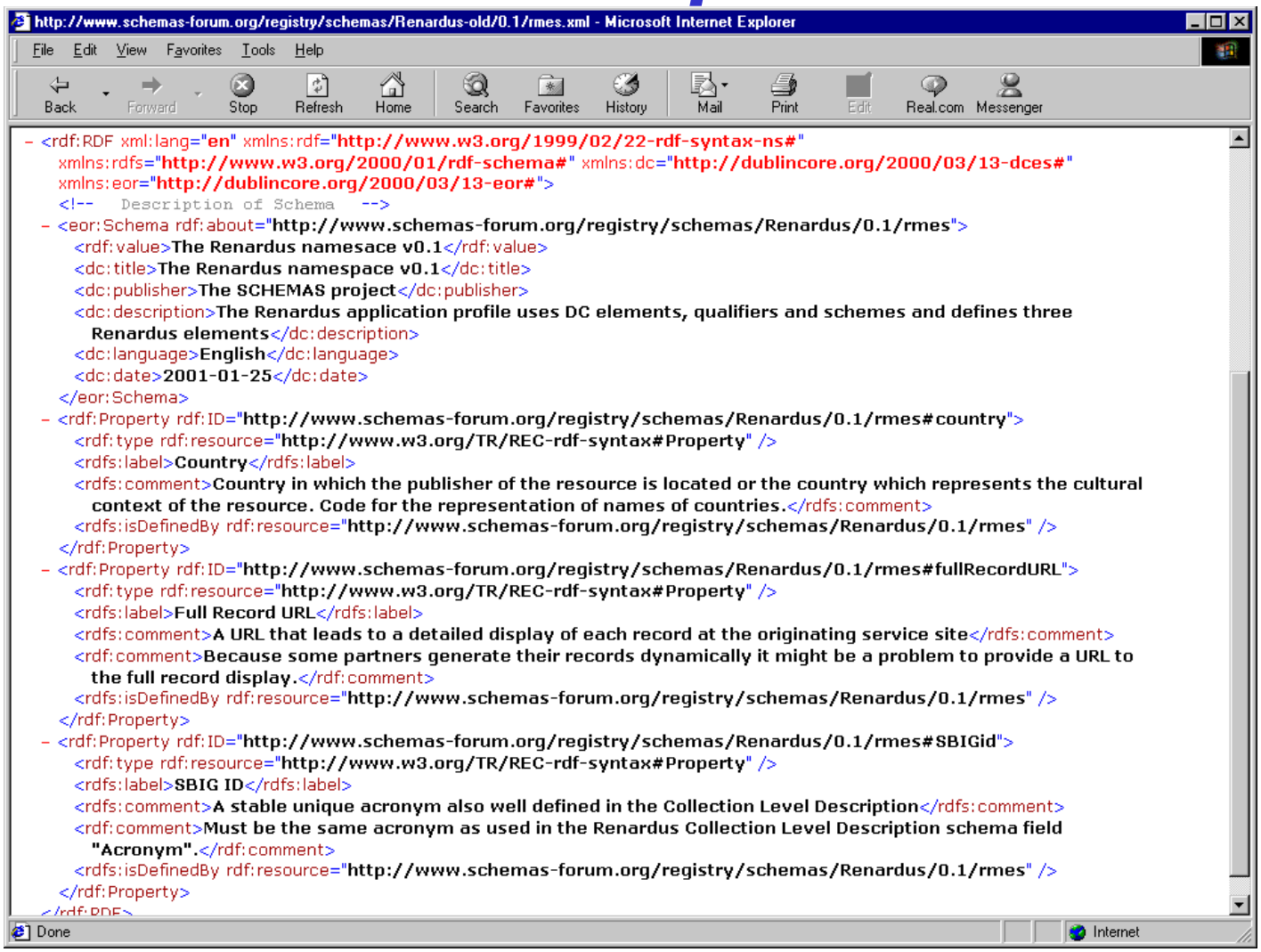
```
<rdf:RDF
  xmlns:rdf = "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs = "http://www.w3.org/2000/01/rdf-schema#"
  xmlns:dc = "http://dublincore.org/2000/03/13-dces#"
  xmlns:dcq = "http://dublincore.org/2000/03/13-dcq#”>
```

Example: Renardus namespace

Define a new term and associated semantics:

```
<rdf:Property rdf:ID=
  "http://www.schemas-forum.org/registry/schemas/Renardus/0.1/rmes#country">
  <rdf:type rdf:resource="http://www.w3.org/TR/REC-rdf-syntax#Property"/>
  <rdfs:label>Country</rdfs:label>
  <rdfs:comment>Country in which the publisher of the resource is located or the
     country which represents the cultural context of the resource. Code for
     the representation of names of countries.
  </rdfs:comment>
  <rdfs:isDefinedBy rdf:resource =
        "http://www.schemas-forum.org/registry/schemas/Renardus/0.1/rmes"/>
</rdf:Property>
```

Renardus namespace



Example: DC-Education namespace

Define a new qualifier to a predefined term:

<rdf:Property rdf:ID="**http://dublincore.org/2000/08/22-dced#mediator**">

<rdfs:label>**Mediator**</rdfs:label>

<rdfs:comment>

An entity that mediates access to the resource. Comment: The audience for a resource in the education/training domain are of two basic classes: (1) an utimate beneficiary of the resource (usually a student or trainee), and (2) frequently, an entity that mediates access to the resource (usually a teacher or trainor). The Mediator element refinement represents the second of these two classes.

</rdfs:comment>

<rdfs:**subPropertyOf** rdf:resource=

"**http://dublincore.org/2000/08/22-dced#audience**" />

</rdf:Property>

SCHEMAS AP format

- Still under development
- Machine processible (RDFS)
- Flat structure -can capture a flat list of elements, but not (yet) complex data models
- New terms within the EOR vocabulary:
  - uses in order to declare reuse of predefined vocabulary terms
  - comment in order to provide local usage guidelines

Example: Renardus AP

…declare usage of elements from multiple namespaces:

```
<!-- semantics from the DCMES, dc:title and dcq:alternative-->
<eor:uses
  rdf:resource = "http://dublincore.org/2000/03/13-dces#title" />
<eor:uses
  rdf:resource = "http://dublincore.org/2000/03/13-dcq#alternative" />
```

Example: Renardus AP

…adapt the definition for dc:creator:

Dublin Core namespace:

Creator: An entity primarily responsible for making the content of the resource

Renardus application profile:

<eor:uses>

<rdf:Description about="**http://dublincore.org/2000/03/13-dces#creator**">

<**rdfs:comment**>

Creator(s) are person(s) which are responsible for the intellectual content of the document(s), e.g. webmasters are not creators.

</**rdfs:comment**>

</rdf:Description>

</eor:uses>

Example: Renardus AP

…declare local usage guidelines for an existing element:

```
<eor:uses>
  <rdf:Description rdf:about =
      "http://dublincore.org/2000/03/13-dces#creator">
   <eor:comment>
     If this field is applicable it is strongly recommended to provide the creator.
     For Renardus normalization process it is strongly recommended that last
     name and first name are clearly distinguishable.
   </eor:comment>
  </rdf:Description>
</eor:uses>
```

Example: Renardus AP

…declare reuse of controlled vocabularies:

```
<!-- dc:subject encoding scheme and associated classes -->
  <eor:uses rdf:resource =
        "http://dublincore.org/2000/03/13-dcq#SubjectScheme" />
  <eor:uses rdf:resource="http://dublincore.org/2000/03/13-dcq#LCSH" />
  <eor:uses rdf:resource="http://dublincore.org/2000/03/13-dcq#MESH" />
  <eor:uses rdf:resource="http://dublincore.org/2000/03/13-dcq#DDC" />
  <eor:uses rdf:resource="http://dublincore.org/2000/03/13-dcq#LCC" />
  <eor:uses rdf:resource="http://dublincore.org/2000/03/13-dcq#UDC" />
```

Example: Renardus AP

…mandate a scheme with a specific element:

```
<!-- mandate a DC subject scheme to be used with dc:subject -->
<eor:uses>
  <rdf:Description rdf:about = "http://dublincore.org/2000/03/13-dces#subject">
   <rdfs:range
        rdf:resource="http://dublincore.org/2000/03/13-dcq#SubjectScheme" />
   <rdfs:domain
         rdf:resource="http://dublincore.org/2000/03/13-dces#subject" />
  </rdf:Description>
</eor:uses>
```

Renardus application profile

Conclusions

- Current SCHEMAS AP format meets requirements of simple discovery
- Limitations
  - Difficulties in capturing complex data models
  - Cannot readily model “obligation”, cardinality and encoding schemes (yet)
- Use of RDF to enhance cross-domain interoperability, as well as convergence and harmonisation within specific domains